

The Core Math for Machine Learning

Your Foundation for Data Science & AI

Why Learn This "Boring" Math?



Go From User to Creator

Understand **how** ML models work, not just **how to run** them.



Debug and Optimize





Diagnose issues in slow or incorrect models instead of just guessing.



Universal Skill

These concepts apply directly to finance, gaming, and business analytics.

Our 4-Part Journey

-  **Linear Algebra:** The structure and "language" of data.
-  **Descriptive Statistics:** The summary and consistency of data.
-  **Probability Basics:** The logic of prediction and uncertainty.
-  **NumPy:** The essential engine that makes it all fast.

Your ML Toolkit Summary

Concept	Role in ML	Key Takeaway
Linear Algebra	Data Structure (Matrices)	All data (images, text, etc.) must be converted to a matrix.
Statistics	Data Cleaning & Summary	Use Mean/Median to find outliers, Std Dev for feature scaling.
Probability	Prediction Logic (Bayes)	Every model output is a probability, not a certainty.
NumPy	Execution Engine	Provides the necessary speed to process large datasets.

Section 1: Linear Algebra

The Structure and 'Language' of Data

Why Learn Linear Algebra?

The ML Core

- Every piece of data (images, text, audio) is converted into a vector or matrix for the computer to process.
- Neural network layers are fundamentally matrix multiplications (transformations).

Real-World Usages

- **3D Graphics & Gaming:** Rotation, scaling, and movement in 3D space.
- **Image Processing:** Applying filters (a matrix convolution).



Data Containers: Vectors & Matrices

Vector (1D)

A shopping list. In ML, a single data sample's features.

e.g., `[Age, Income, Height]`


Matrix (2D)

The entire dataset. Rows = Samples, Columns = Features.

e.g., `[[25, 50k, 170], [35, 80k, 185]]`

Scenario Exercise: The Bakery Order

 **Goal:** Calculate total ingredients using Scalar Multiplication & Vector Addition.

 **Choco Recipe:** `[2 flour, 1 sugar, 0.5 chocolate, 0 oats]`

 **Oatmeal Recipe:** `[1 flour, 0.5 sugar, 0 chocolate, 1 oats]`

 **Order:** 10 Choco batches + 5 Oatmeal batches.

 **Question:** What is the total vector of ingredients?

Solution: The Bakery Order

Code

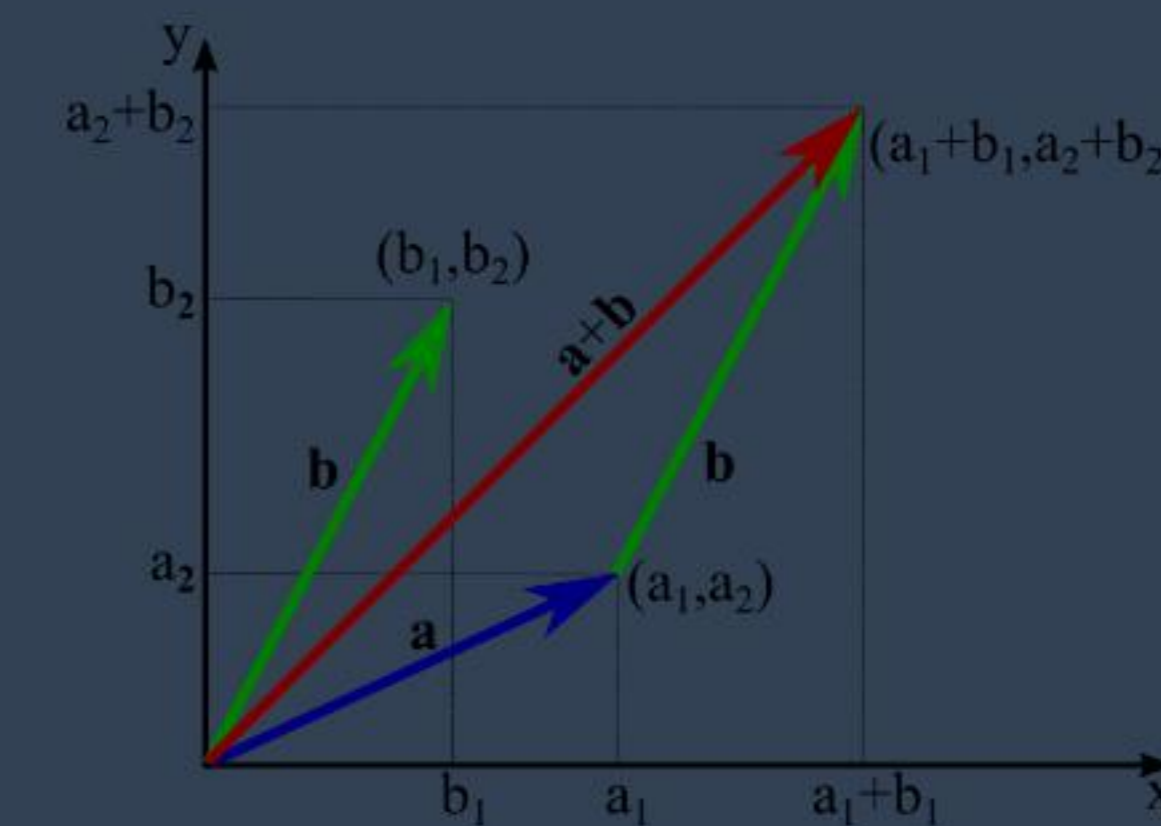
```
import numpy as np
choco = np.array([2, 1, 0.5, 0])
oatmeal = np.array([1, 0.5, 0, 1])
# 1. Scale
order_choco = choco * 10
order_oatmeal = oatmeal * 5
# 2. Add
total = order_choco + order_oatmeal
# [20, 10, 5, 0] + [5, 2.5, 0, 5]
```

Answer

The total ingredient vector is:

[25, 12.5, 5, 5]

(25 flour, 12.5 sugar, 5 chocolate, 5 oats)



Transformations (The "Selfie Filter")

Definition

A matrix that moves, stretches, or compresses a vector. Think of a "dog ears" selfie filter - it's a matrix that transforms your face's coordinates.

ML Application

"Learning" is finding the optimal **weights matrix** that successfully transforms input data (e.g., pixel values) into the desired output (e.g., the label "cat").

Section 2: Descriptive Statistics

Summarizing Data: Central Tendency & Dispersion

Why Learn Statistics?

The "Why"

- ****EDA:**** Essential for Exploratory Data Analysis. Helps you detect errors, bias, and outliers *before* training.
- ****Model Prep:**** Techniques like Standardization are used to prepare features for training.

Real-World Usages

- ****Finance:**** Calculating volatility (risk) using Standard Deviation.
- ****Quality Control:**** Measuring process consistency.



Key Measures: Central Tendency

Mean (Average)

The "Average Joe." What everyone gets if the total is split evenly.

Warning: Fooled by outliers (Bill Gates walks in).

Median (Middle)

The "Middle Person." The value in the dead center after sorting.

Benefit: NOT fooled by outliers.

Key Measures: Dispersion

Standard Deviation

The "Drama Queen" number. Measures the *average distance* of every point from the mean. Low = consistent, High = inconsistent.

Purpose in ML

Spotting Outliers: When Mean \neq Median.
Feature Scaling: Normalizing features (like Age vs. Salary) so the model treats them fairly.

Scenario Exercise: Coffee Shop Consistency



Quick-Cup: `[3, 4, 3, 5, 4, 2, 3, 4]`



Slow-Sip: `[1, 8, 2, 9, 1, 7, 10, 4]`

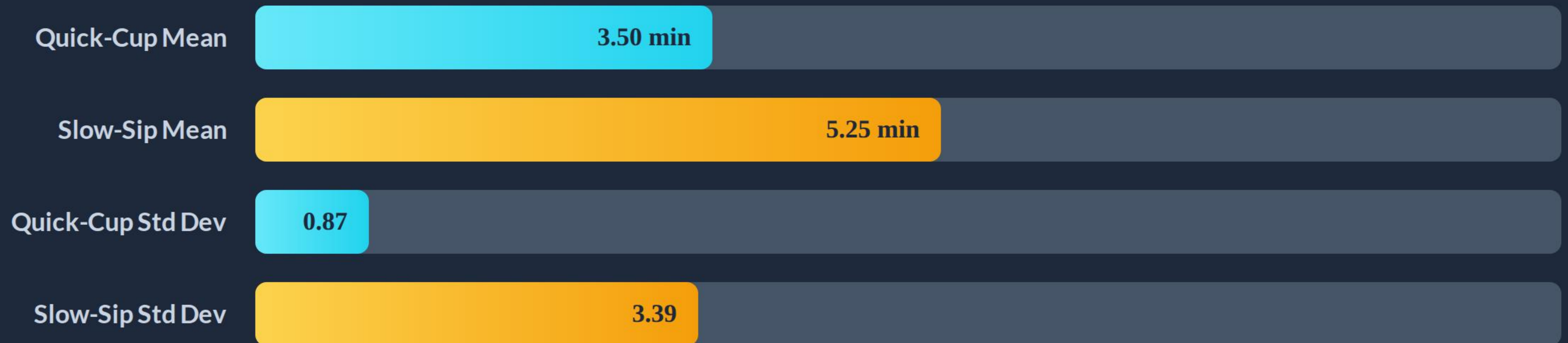


Question 1: Which shop is **faster** on average? (Check Central Tendency)



Question 2: Which shop is **more reliable**? (Check Dispersion)

Solution: Coffee Shop Consistency



*****Answer:**** **Quick-Cup** is faster (lower mean) and far more reliable (lower std dev). Slow-Sip is inconsistent.*

Section 3: Probability Basics

The Logic of Prediction and Uncertainty

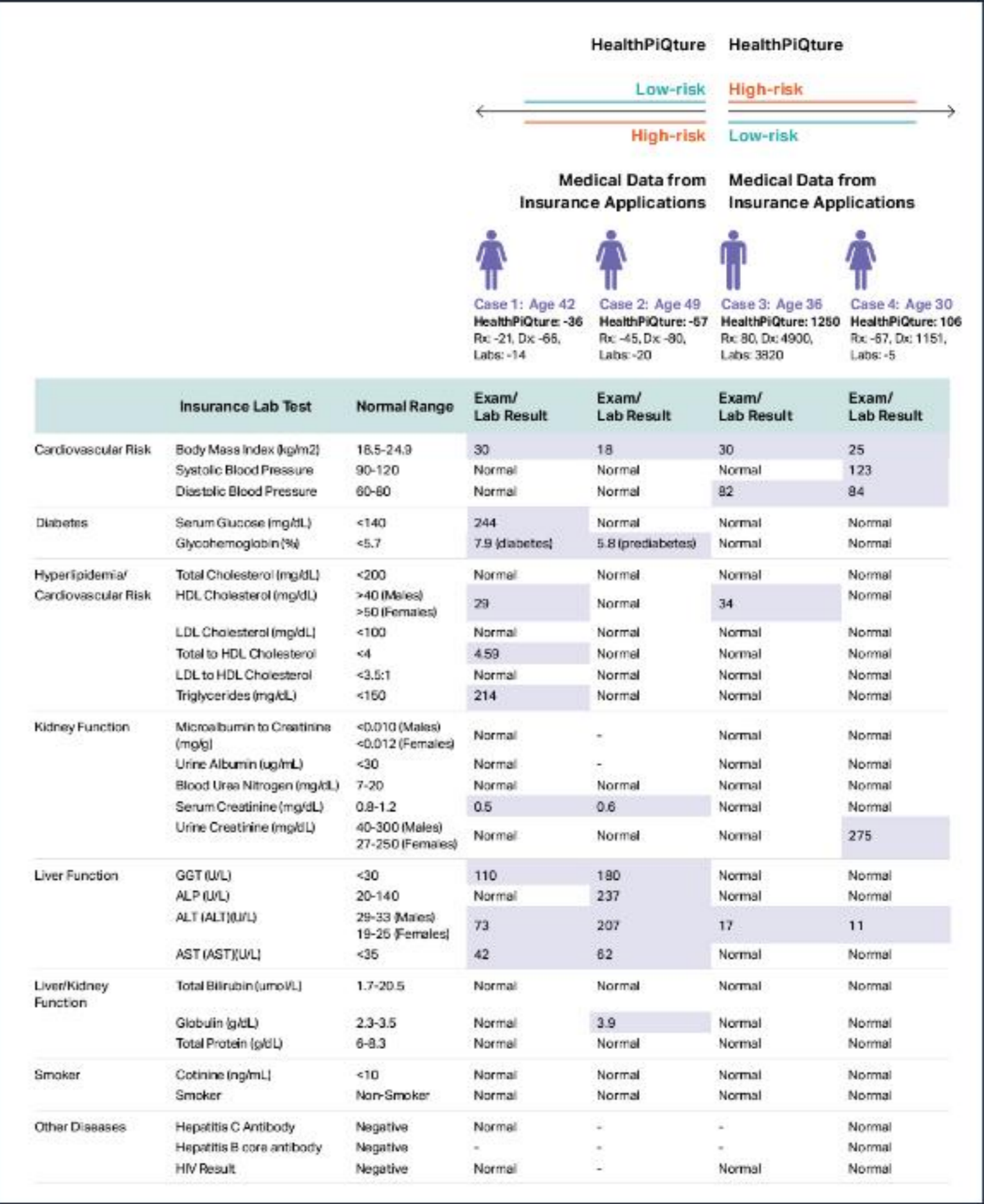
Why Learn Probability?

The "Why"

- **The ML Output:** Every prediction is a probability ("95% chance this is a dog"), not a 100% certainty.
- **Classification Logic:** Building spam filters and medical diagnostics relies on Bayes' Theorem.

Real-World Usages

- **Insurance:** Calculating risk profiles.
- **Medical Testing:** Understanding false positive rates.



Conditional Probability

Definition: $P(A | B)$

The probability of A... **given that** B... has already happened.

e.g., $P(\text{Umbrella} | \text{Raining})$

(The chance you have an umbrella *given that* it's raining)

ML Example

$P(\text{Spam} | \text{"win" was in email})$

(The probability an email is spam *given that* it contains the word "win")

Bayes' Theorem: The Spam Filter Logic



1. Base Rate (10k Emails)

100 are Spam (1%)
9,900 are Ham (99%)



2. Find Clue ("money")

50% of Spam (50 emails)
2% of Ham (198 emails)



3. New Pool

Total "money" emails:
 $50 + 198 = 248$



4. Final Answer

$P(\text{Spam} \mid \text{"money"})$
 $= 50 / 248 \approx 20.2\%$

Bayes' Theorem: The "Aha!" Moment

“

The probability is low because 'Ham' (non-spam) was so much more common to begin with, creating more false positives.

”

Section 4: NumPy

The Essential Engine for Speed

Why Learn NumPy?

The "Why"

- ****Speed (Vectorization):**** Performs math in optimized C code, 100x-1000x faster than Python `for` loops.
- ****The Standard:**** The foundational library for all other tools like Pandas, Scikit-learn, and TensorFlow.

 Python and NumPy logos

NumPy vs. Python List

Python List

A clumsy filing cabinet. Can hold anything (numbers, strings).
Flexible, but ****VERY SLOW**** to do math.

NumPy Array

A military ammo box. Only holds one type (e.g., numbers).
Rigid, which makes it ****BLAZINGLY FAST****.

Conclusion: Your ML Toolkit

 **Linear Algebra** is the ****Structure**** to store data.

 **Statistics** is the ****Understanding**** of that data.

 **Probability** is the ****Logic**** to make predictions.

 **NumPy** is the ****Engine**** to make it all work fast.

Questions?

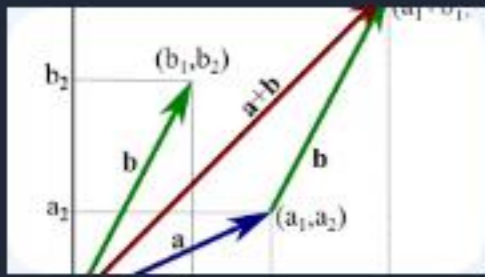
Thank you for your attention.

Image Sources



https://img.freepik.com/premium-photo/machine-learning-stock-ai-deep-learning-blockchain-neural-network-concept-brain-visualization-with-shining-wireframe-net-graphic-abstract-background-3d-render_1257429-92497.jpg

Source: www.freepik.com



https://mathinsight.org/media/image/image/vector_2d_add.png

Source: mathinsight.org



<https://cdn.boldbi.com/wp/pages/dashboards/finance/financial-mgmt.png>

Source: www.boldbi.com



https://www.partnerre.com/wp-content/uploads/2022/09/table_3.png

Source: www.partnerre.com



<https://i.sstatic.net/dDepR.jpg>

Source: stackoverflow.com